(12) INTERNATIONAL APPLICATION PUBLISHED UNDER THE PATENT COOPERATION TREATY (PCT)

(19) World Intellectual Property Organization
International Bureau

(43) International Publication Date
19 July 2001 (19.07.2001)

PCT

(10) International Publication Number
WO 01/52510 A1

(51) International Patent Classification[7]: H04M 3/36, 3/51

(21) International Application Number: PCT/GB01/00129

(22) International Filing Date: 12 January 2001 (12.01.2001)

(25) Filing Language: English

(26) Publication Language: English

(30) Priority Data:
0000735.1 13 January 2000 (13.01.2000) GB

(71) Applicant *(for all designated States except US)*: **EYRE-TEL PLC** [GB/GB]; Kings Court, Kingston Road, Leatherhead, Surrey KT22 7SZ (GB).

(72) Inventor; and

(75) Inventor/Applicant *(for US only)*: **BLAIR, Christopher, Douglas** [GB/GB]; Ivor Cottages, Balneath Farm, South Chailey, East Sussex BN8 4AP (GB).

(74) Agent: **GILL, David, Alan**; W.P. Thompson & Co., Celcon House, 289-293 High Holborn, London WC1V 7HU (US).

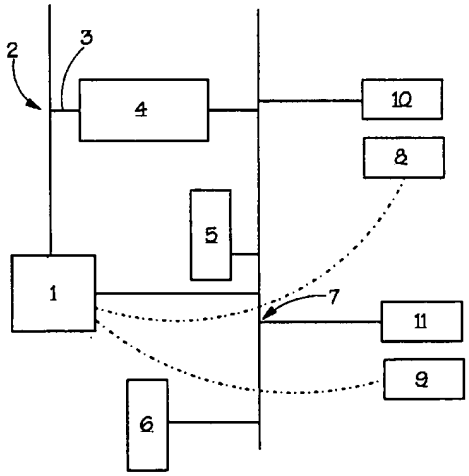
(81) Designated States *(national)*: AE, AG, AL, AM, AT, AU, AZ, BA, BB, BG, BR, BY, BZ, CA, CH, CN, CR, CU, CZ, DE, DK, DM, DZ, EE, ES, FI, GB, GD, GE, GH, GM, HR, HU, ID, IL, IN, IS, JP, KE, KG, KP, KR, KZ, LC, LK, LR, LS, LT, LU, LV, MA, MD, MG, MK, MN, MW, MX, MZ, NO, NZ, PL, PT, RO, RU, SD, SE, SG, SI, SK, SL, TJ, TM, TR, TT, TZ, UA, UG, US, UZ, VN, YU, ZA, ZW.

(84) Designated States *(regional)*: ARIPO patent (GH, GM, KE, LS, MW, MZ, SD, SL, SZ, TZ, UG, ZW), Eurasian patent (AM, AZ, BY, KG, KZ, MD, RU, TJ, TM), European patent (AT, BE, CH, CY, DE, DK, ES, FI, FR, GB, GR, IE, IT, LU, MC, NL, PT, SE, TR), OAPI patent (BF, BJ, CF, CG, CI, CM, GA, GN, GW, ML, MR, NE, SN, TD, TG).

Published:
— *with international search report*

For two-letter codes and other abbreviations, refer to the "Guidance Notes on Codes and Abbreviations" appearing at the beginning of each regular issue of the PCT Gazette.

(54) Title: SYSTEM AND METHOD FOR RECORDING VOICE AND THE DATA ENTERED BY A CALL CENTER AGENT AND RETRIEVAL OF THESE COMMUNICATION STREAMS FOR ANALYSIS OR CORRECTION



(57) Abstract: The invention provides for a communications recording and analysis system including means for recording one or more communication streams, means for identifying the recorded stream, means for retrieving the content of said recordings by identifier tags, and wherein additional real-time information is inferred from analysis, in real-time or later, of keystrokes entered at a computer/terminal handling the interaction, and/or computer mouse actions, and/or internet traffic emanating from, or terminating at, any one or more of a number of computers/terminals handling the interaction, and/or the words and/or prosody spoken during the interaction is recorded. Furthermore graphical display means are provided such that the presentation of call flow recording is in the form ofa direct graph showing the progress of the calls through the various states and transitions.

WO 01/52510 A1

SYSTEM AND METHOD FOR RECORDING VOICE AND THE DATA ENTERED BY A CALL CENTER AGENT AND RETRIEVAL OF THESE COMMUNICATION STREAMS FOR ANALYSIS OR CORRECTION

The present invention relates to a system and method for analysing communication streams.

Currently many commercial entities perform substantial amounts of their business via telephone or Internet contact with their customers. The analysis of such contact can therefore help businesses to improve the quality and efficiency of their services, and assist with customer retention and, ultimately, profitability. Attempts have been made previously to achieve such analysis in a satisfactory manner and to a satisfactory degree. For example, many businesses have, for some time, recorded some of their communications streams e.g. telephone calls between their staff and their customers. Traditionally this was done to satisfy regulatory requirements or to help resolve disputes. More recently, the emphasis has moved towards the reviewing of these interactions from a quality perspective. This is intended to identify good and bad aspects of particular calls with a view to improving the level of customer service given. Recently, for example, recording the activity on a PC screen has been undertaken to improve the completeness of the review procedure with the reviewer able to see how accurately staff are entering information received via the telephone.

Also, it has been known to employ Call Detail Recording (CDR) systems to prevent perceived abuse of telephone systems and to apportion costs to the department or individual making the calls.

Originally such records were printed out directly from the Private Automatic Branch Exchange (PABX) onto a line printer. Later, systems were designed to store this information in a database allowing more sophisticated

reporting and searching for calls on the basis of one or more of the stored call details. More recently, Computer Telephony Integration (CTI) interfaces have been provided that give this information in real-time, during the call.

Further, several systems currently exist that use call recording in combination with CDR or CTI and a database application in order to perform routine monitoring of calls with the intention of identifying weaknesses in individual Customer Service Representatives (CSRs). Typically a small percentage of the CSR calls are reviewed and "scored" against a set of predetermined criteria to give an indication of the quality of that particular member of staff.

Within a call-centre environment, it should also be noted that rather than simply using standard PC office automation applications when dealing with customers, staff in most call centres use increasingly sophisticated applications that help them to handle the calls more efficiently and effectively. Helpdesk applications and telemarketing call scripting applications are examples of such applications. Most such applications share the following characteristics:

(i) they guide the staff member through the interaction, either explicitly suggesting questions to ask or implicitly e.g. by presenting a form with various fields to be filled in with details to be gleaned from the customer; and

(ii) the progress of the call is, to some degree, dictated by the answers given during the call.

As an example, a service call to a help desk may flow through the following stages:

(i) take customer details and verify maintenance contract is up to date;

(ii) take details of the problem being experienced;

(iii) attempt to resolve using appropriate set of

diagnostic questions;

(iv) if not solved, agree schedule for engineer to visit; and

(v) give customer a problem reference number.

Within each of the above main stages there will be one or more sub-stages e.g. (i)(a) caller's name (i)(b) zip code etc.

Current systems employed for CSR quality monitoring require the manual review of calls in order to determine a quality "score". This prohibits the review of more than a few percent of all calls and the accuracy of the human scoring process and its objectivity are at best questionable. When coupled with a small sample size of typically only a few calls per CSR per week, the resultant scores offer a limited degree of quality monitoring. Also there are only a few aspects of the call that allow for automated scoring. The most common such aspect is the call duration which, for a known telemarketing script, should be within a known range. A portion of the overall score (perhaps 5%) may be allocated to such an aspect of the call.

Call centre management seeks to handle calls as efficiently as possible yet take advantage of cross-selling and up-selling opportunities whilst the customer is already on the line. To this end, the way in which calls are handled within the call centre is of critical importance to the overall efficiency and profitability of the operation. As the sophistication of call handling systems and processes increases, so the complexity of the call flow designs increases such that it is very difficult to fully test and exercise these "scripts" prior to, or even during, their full-scale deployment. Sophisticated scripts may require a hundred pages or more of documentation including the "call flow diagram" and the associated questions and forms presented to the CSR. With many hundreds or even thousands

of possible paths through such a script, any one CSR will not have sufficient experience of many particular routes through the script to allow for the objective evaluation of the effectiveness or otherwise of, for example, the suggested way to handle a particular objection raised during a sale.

Current systems typically store details of each call in a relation database typically with one record of more or less fixed structure per call. These details may include indications of the type of call, the "completion code", or whether a particular option was chosen or not etc. More advanced systems cater for the cases where a single "call" actually contains more than one transaction (e.g. request a quotation for two different insurance policies). This is typically done by subdividing the call into two or more parts each of which then has a database record associated with it.

Existing systems use one or more database records per call and the identification of calls normally requires that the user determine in advance which aspects of the call's progress are to be stored. Timing information stored is often limited to total call duration. The relative time between different points in the call is difficult to determine. Calls that "loop" through the same part of the script several times are not well described by a single database record.

It has also been appreciated that customers and potential customers generally much prefer a "warm call" to a "cold call" i.e. a call where the caller obviously has taken the trouble to find out about a customer's previous dealings with their business rather than merely selecting the name from a list without any knowledge of the particular customer's details; many of which may already have been given previously.

Many businesses now proactively call their customers on a regular basis to try and "cross-sell" or "up-sell" with other products that they have determined may be of interest to the customer.

However, currently, where a follow-up call is to be made as the result of a previous call perhaps made several months earlier there is often little information available to the CSR responsible for the follow-up call. For example, a routine call to all mortgage customers may include the question "Have you reviewed your life insurance cover recently"?. According to the answer given, the customer may be placed on a list for a follow up call the next time a new life insurance policy is launched. At the time of the original call, the CSR will likely do no more than tick a "Yes/No/Maybe" option.

When the customer is called some months later, all that the new agent knows is that the customer previously said they may have been interested and the factual details presented about the customer's previous purchasing history. They do not know exactly how or in what tone of voice the customer responded several months earlier.

Additionally, such systems are open to abuse by the CSRs as they may be rewarded, or at least measured, by the number of such follow-up leads they record. Hence there is a temptation for anything other than a point-blank "No" to be recorded as a "Maybe".

It is also acknowledged that, in order to improve business processes, train CSRs and identify problems, there is a desire to identify individual calls or sets of calls which are handled particularly well or badly or that highlight deficiencies, opportunities, trends or anomalies.

Existing systems limit the selection criteria to those that

can be derived solely from call details. These use individual database records associated with each call to store data fields about each call. These are particularly inflexible and need to be designed prior to the query. For example, if a user identifies that they will want to search for calls where the "Help" key was pressed, they could tag each call with a binary flag and set this to true as the help key is pressed. A more sophisticated solution would be to use a time field and store the time at which the help key was pressed. However, if the user then wants to find calls where "Help" was pressed twice or more, this known previous database scheme proves inadequate.

When considering call-replay, and irrespective of the reason for wanting to replay a specific call, there is often a desire to listen to a particular section of a call rather than the whole call. For example, an insurance claims handler will want to confirm that a driver now known to have a previous conviction disclosed this when asked during the initial policy quotation. Being able to jump directly to the portion of the call where the customer was asked this question avoids the need to browse through the whole call. Alternatively, a mail-order company CSR needing to correct an invalid address would like to jump to the part of the original call where the address was given.

Current systems require users to either listen to the whole call or to jump around the call listening to snippets until they "home in" on the required section. This is time consuming and prone to error.

Further it has long been possible to perform automated speech recognition in an attempt to transcribe or at least to highlight key words within telephone conversations. The capabilities of such systems (accuracy, tolerance of natural conversation speed, cost effectiveness) have been gradually improving during recent years.

To obtain maximum accuracy, currently employed algorithms are very CPU and memory intensive. It is unlikely that many customers will be able to afford to deploy full recognition across all conversations they record.

Simply applying generic recognition to the whole of a conversation requires the recognition algorithm to infer the context and grammar from the conversation in order to refine its decisions as to which words it heard. This process is CPU intensive and less accurate than being explicitly told the current context of the conversation.

It will therefore be appreciated that the current schemes of communication analysis are disadvantageously limited.

The present invention seeks to provide a system and method for analysing communication streams having advantages over known such systems and methods.

According to one aspect of the present invention there is provided a system for communications recording and analysis including means for recording one or more communication streams, means for identifying the recorded streams, means for retrieving the content of said recordings by identifier tags and wherein additional real-time information relating to the progress of the said communication streams is recorded.

Preferably, the communication streams and associated progress streams are linked by means of a cross reference within respective call detail records in a database of recordings.

In one embodiment, the communication streams can be interleaved with the call flow recordings and a composite stream is recorded.

Advantageously, the progress information may be inferred from analysis, in real-time or later, of keystrokes entered at a computer/terminal handling the interaction.

Alternatively, the progress information may be inferred from analysis, in real-time or later, of computer mouse actions.

Yet further, the progress information may be inferred from analysis, in real-time or later, of internet traffic emanating from, or terminating at, any of a number of computers/terminals handling the interaction.

Additionally, the progress information may be inferred from analysis, in real-time or later, of the words and/or prosody spoken during the interaction.

It should be appreciated that the contents of the call flow data stream may be used to refine the recording and/or analysis of the main stream(s).

Preferably, retrieval of specific sets of recordings may be performed by selection from the call detail records describing each of the recordings and/or the presence, repeated presence or absence of specified events or call states.

Advantageously, the criteria may include relative or absolute timing information.

Further, the presentation of the content of individual recordings may be in the form of a graphical display including bars and/or pictures and wherein the location, colour and/or size of which may be determined by the recorded call flow recording.

In particular, the presentation of one or more call flow recordings may be in the form of a directed graph showing

the progress of the calls through the various states and transitions.

Also, the position, colour, labelling, texture and other graphical attributes of the nodes and/or lines can be determined by the number of calls following that route; the time they spend in the various states and/or other attributes derivable from the call flow recordings.

Still further, the display can be populated, and the graphical attributes modified, in real-time to reflect the timing information recorded within the call flow recordings.

In particular, the speed of replay can be varied, for example, including "paused", "forwarded" or "reversed" at any rate including "single-step" mode.

Also, the set of call flow recordings may be refined by selecting, for example, via a double-click, a specific transition as discussed further later.

Advantageously, specific nodes and/or transitions can be highlighted according to automatically or manually defined rules so as to draw attention to the presence or absence of particular call flow routes.

The invention is advantageous in that the ability to track the progress of the call through the various stages allows more criteria to be subsequently scored automatically and hence objectively. Since computer-scoring can be easily and cost effectively achieved, it can be performed economically on a much larger sample of the calls - often the full 100%. For example, the time spent taking customer details should be within allowed ranges. Calls that did not result in the sale of product "X" should not have spent more than t seconds trying to sell this product before moving on to other propositions. Calls should not occur where the

customer had to go back and re-enter details that should have been picked up earlier. Such a system allows perhaps 20-30% of the overall quality "score" to be determined automatically and therefore objectively and across all calls. This gives much better statistical significance with less manpower.

With regard to call flow design and call centre management, the invention can for example:

present N examples of calls following a specific path through the call flow script e.g. a rarely exercised branch; and

further filter these calls according to more traditional call detail records e.g. "Service calls for Model XYZ that ended without problem resolution"

Having selected the set of calls of interest, these are then presented to the user and this can allow them to:

review just a specified section of the call (e.g. answer to question 143) without having to dip into the whole call to determine the appropriate part of the call;

review the whole call so as to hear the required section in context; and

see the timing information associated with the calls e.g. how long spent in the specified section; how long between start of sales pitch and this section etc.

Turning now to the personalisation of calls, the invention can advantageously allow the few seconds of response to the relevant question to be retrieved efficiently without the user having to listen to the rest of the call. Using this feature, two strategies can be adopted. First, all of the responses to the question can be reviewed by a dedicated individual or team prior to the campaign getting underway.

As they listen to all of the responses, they can (a) prioritise the customers from most to least likely to buy and hence choose which ones will eventually be called and (b) ensure that the script that is to be used when the campaign starts includes answers to the issues likely to be raised. Hence they can improve the effectiveness of the campaign and reduce the number of wasted calls that will be made. Note that feature (a) above requires all responses to be listened to, whilst feature (b) only requires that a statistically significant sample be reviewed.

Secondly, during the campaign, agents may be presented with the previous response either in audio form or in textual form (manually or automatically transcribed). The former gives them the tonal and emotional content of the response but is less practical where autodiallers are used to present calls to CSRs as soon as the customer answers. Either approach lets them speak to the customer from a position of advantage, knowing exactly how they reacted to the question when last asked. Presenting the previous response first would allow the CSR to determine whether it is worth placing the call or not.

By recording the detailed progress of the call along with the call details and call content, the system advantageously allows the user to identify sets of "interesting" calls. Such calls might be those exhibiting the presence or absence of certain events within the call e.g. CSR requested "help" during the quotation process. This could identify CSR specific training needs or areas where all staff experience difficulty that could be improved.

Alternatively, or in addition, where one product was purchased but details of others were also given, it might be worth calling the customer soon to try again to sell them these other products.

Also, where the CSR had to return to a previous part of the script, the reason for this can be considered. For example, was a question ambiguous, a mistake made or did the customer change their mind?

Further, where less than 5 seconds was spent in the "confirmation" screen where customers should have the small print explained to them, this could suggest that the CSR was rushing on to the next call too fast.

Of course, poor typing skills might be indicated by excessive corrections and slow data entry.

Efficient replay of calls can advantageously be achieved by recording the detailed progress of the call along with the call details and call content so that the system allows the whole call to be shown on a Graphical User Interface (GUI) with graphical indications of the current state of the call and significant events within the call. The user can then choose which section(s) of the call to play by clicking on the appropriately marked points. Also, required segments of the call to be presented in isolation and played individually or one after the other.

The present system, through monitoring and recording the progress of the call along with the audio content of the call, advantageously allows speech recognition algorithms to be directed at only those sections of conversations where maximum value is likely to be obtained. For example, no interpretation of the speech is undertaken whilst the customer's address is being taken. Likewise, interpretation can be concentrated on sections where sales objections are being handled or likely to be handled. Further, the algorithms can be directed in their selection of likely words according to the context of the conversation. For example, if a CSR has tabbed to the "Destination" field on screen, the customer is therefore more likely to have said

"Paris" than "Pairs". Also, the algorithm can be directed as to the likely speaker - whether CSR or customer. That is, where independent transmit and receive audio streams are not available, the progress through the call can be used to identify which speaker is most likely to be talking.

The capabilities described above also allow the system to be used for the following additional and advantageous purposes. Usability analysis can determine how many keystrokes/mouse clicks were required to perform the most common functions and also identify the most common functions. Testing and Verification can also be achieved and used to find any calls where an order was taken that had not previously taken the customer through the required explanatory text thereby identifying an invalid path through the call flow script. CSR Quality Monitoring allows for the identification of those CSRs that spend more, or less, time than expected in specific sections of the call flow process or that follow certain paths through the call flow more or less often than the norm.

The invention is described further hereinafter, by way of example only, with reference to the accompanying drawings in which:

Fig. 1 is a block diagram of a recording and analysis system embodying the present invention;

Fig. 2 is a functional block diagram of a recorder such as that in Fig. 1;

Fig. 3 is a functional block diagram of an Application Programming Interface (API) for use within an embodiment of the present invention;

Fig. 4 is an illustration of a graphical representation of the progress of a call;

Fig. 5 is a flow diagram such as that displayed in accordance with an embodiment of the present invention to illustrate call progress; and

Fig. 6 is a further flow diagram derivable from the

diagram of Fig. 5.

Turning first to Fig. 1, one or more datastream recorders 4 are connected to speech transmission circuits 2 which in turn are typically connected to a Private Automatic Branch Exchange (PABX) or similar telephony switching system 1. Connection is achieved by means of a high impedance tap 3 which does not impact the normal use of the speech paths but also allows the recorder to monitor the signals on the paths.

The recorders 4 are also connected to a local area network such as an ethernet (7) over which they communicate with other components of the system and can receive data for storage along with the voice data that they are recording from the speech paths (3). This data can be provided by any application on the network such as central business applications running on servers 6 or on an end user's desktop 10.

A server 5 consolidates details of all calls recorded by the recorders 4 which may be scattered across a local or wide area network and maintains a central database of "call records" which can be searched using standard SQL techniques. This also maintains details of the current location of removable media and the calls they contain.

Also running on a server 5 as part of the recording system is typically an application, for example UNIFY, which interprets Computer Telephony Integration (CTI) data from the telephony switch 1. This information is used to control the recorders 4 and indicate when to start and stop recording on both the speech and data channels. This information is also used to "tag" the calls with information regarding the call, such as which extension 8 the call was directed to. The recording of the data streams can also be controlled by this application and these too can be tagged

with additional detail, such as which desktop 10 they relate to. Additionally, data stream recordings can be tagged with the identifier of the speech call that is in progress on the same desktop. For example, data from desktop 10 would be tagged with the identifier of the extension 8 on that desktop.

The voice and data calls that have been recorded can be retrieved by applications running on PCs on the network 11. These applications search the call-details database held on the server 5 to determine which call(s) they wish to replay and where these calls are currently held. The call content is then requested from the recorder holding the disk, tape or optical media on which it has been stored.

The following components of an overall system are currently available and are used as an underlying platform on which the specific enhancements used to provide the benefits described above can be deployed.

Multimedia Stream Recorders

The recorder (4) is an example of a multi-channel voice recorder capable of storing up to 128 conversations simultaneously. Fig. 2 shows the relevant details of the recorder. The speech signals are presented via a high impedance tap 12 and are typically compressed by an appropriate line interface card 14. At regular intervals the blocks of data to be stored are passed via the internal data recording interface 16 for indexing and storage to hard disk 18, or optionally, Digital Audio Tape (DAT) media for longer term archive. Calls can be "tagged" with arbitrary data fields describing the call and subsequently retrieved for replay or analysis. In addition to recording voice calls, the recorder can record data streams representing arbitrary communication types. Examples include PC screen capture recordings, messages for display on underground

trains in-cab displays etc. These are typically presented as Internet Protocol (IP) packets via an ethernet cable 13 connected to an appropriate Network Interface Card (NIC) 15. The contents of these packets are passed through the same internal storage API 16 for storage to disk.

Computer Telephony Integration (CTI) Control and Tagging.

The UNIFY component mentioned above is used to interpret one or more data streams typically provided by telephony switches, Automatic Call Distributors (ACDs) or applications within a call centre. By passing the information flows from these and applying customers defined rules, the UNIFY component will control the recorders to start, stop, pause, resume or break recordings on specified voice and/or data channels and/or to "tag" recordings, current or past, with specified data. These data fields ultimately form part of that recording's call detail record and can later be used to search for it.

Enterprise-wide Recording and Retrieval Services

E-Ware is an example of a suite of NT 4.0 applications which manage one or more recorders to provide, across the customer's local or wide area network, an enterprise-wide set of recording and retrieval services. Application Programming Interfaces (APIs) provide access to recording control, configuration, status monitoring, search and retrieval mechanisms. The system consolidates the call detail records from all the recorders in the system into a central, open relational database allowing queries of arbitrary complexity to be performed using SQL. Additionally, the system records the contents and current location of all removable media to which calls have been recorded.

Data Recording System

One of the APIs of systems such as E-Ware provides for data recording capabilities and can be used by screen capture mechanisms.

The generic data recording API 21 is implemented for example by Eware2DR.DLL on Windows platforms and allows applications 19 to:

- register data streams as being available for storage;
- start and stop recordings on such data streams with the data being transferred across the network 23 via IP packets formed by an IP protocol stack 22 and ultimately stored on the recorders (4);
- provide their own timing information within the data stream supplied or have the system automatically package the data inside a protocol that adds time information at the required level of accuracy and granularity.

Replay Application

Windows applications such as "Replay Studio" can be provided that allow a user to:

- specify search criteria and perform searches to select a required set of call details from the central database;
- view the results of these searches in tabular and/or graphical formats;
- retrieve the contents of selected calls from the recording system for delivery to a local or shared cache area for subsequent replay; and
- replay one or more selected calls in a way appropriate to the type of call. Examples are voice calls played via a soundcard and screen capture calls replayed in an on-screen window.

The application consists of an ActiveX framework into which

additional data visualisation and replay mechanisms can be added to support new call types.

Call Flow Recordings (CFRs)

The particular features related to this embodiment of the invention are as follows. The main purpose of these enhancements is to provide a Call Flow Recording (CFR) that details - to the level of detail required for a given application - the progress of a call through the system. this CFR is not a database record but is actually a recording of a real-time data stream that allows the progress of the call to be reconstructed including both the route it took through the call handling process - potentially down to the individual key-strokes entered - and when each step occurred.

These CFRs are stored within the generic recording system as "calls" of a new, well-known, Format type. This allows the retrieval and replay tools to recognise them and display them appropriately as opposed to trying to replay them as audio calls. The format identifier used is chosen to be in the range reserved for variable bit rate streams.

These CFRs are tied to the other components of the call, such as voice recording and screen content record, by use of cross-reference fields within their call detail records. Each CFRs call detail record includes the globally unique reference number of the "parent" call - typically a voice recording - to which it refers.

Additionally, "Parent" and "Child" flag fields can be used within the call detail records to alert applications to the fact that the voice call in question has one or more associated "calls" and, conversely, that the CFR has a related parent call and should not be viewed in isolation.

An alternative embodiment merges the context or progress information with the main recording so that the recorded data stream is an amalgam of the original plus the state information. A simple packetisation protocol allows the combined stream to be decoded into blocks of original and additional state information. This packetisation protocol may be further extended so as to include many different data streams within the single file. Any one or more of these streams may be extracted from the whole for subsequent analysis or replay.

Explicit Context Notification

Where applications involved in the progress of calls through the call centre are required to explicitly advise the recording system of the call's current state, this can be achieved by using an API which is layered on top of the generic data recording API. This "Call Flow Recording API" (CFR-API) provides the following functionality:

- it advises that the call is entering stage *N*, and offers user definable parameters *P1...Pn* associated with this transition which are typically the name of the stage;
- it advises that the call is leaving stage *N*, and offers user definable parameters *P1 ...Pn* associated with this transition which are typically the name of the stage; and
- it advises that the event *E* that is occurring e.g. a sale being made, and offers user definable parameters *P1 ...Pn* associated with this transition, for example the value of the sale).

It should be noted that several levels of detail can be stored since calls can enter multiple stages without leaving a higher level one. For example, a sales call may generate the following calls to the API as it is handled:

- entering Customer Identification Stage;
  - entering Name Determination Stage;
    - entering Surname Determination Stage;
    - leaving Surname Determination Stage;
    - entering FirstName Determination Stage;
      - entering HelpScreen;
      - leaving HelpScreen;
  - leaving FirstName Determination Stage;
- leaving Name Determination Stage;
- entering Address Determination Stage; and
- leaving Address Determination Stage.

Implicit Context Determination

In some cases it may be that not all of the applications involved in the handling of calls will be enhanced to provide the above-mentioned explicit notification of call progress. In such cases it may be necessary to infer the progress of the call from other sources such as screen capture, network (e.g. `web') traffic and/or speech analysis of the voice records.

In such cases, these other data streams may be used as more or less satisfactory proxies for the missing or incomplete call flow recording. Three examples of how call progress can be inferred are given below.

These inferred call flow elements may be analysed in real time and merged into the overall call flow recording or may be determined at a later date by retrieving the data recording from which they are to be extracted.

First, the progress can be inferred from the key-strokes entered. As the entry of data via the keyboard is such a common and essential part of most call handling, keystrokes can be, by default, recorded as an integral part of the call flow recording for a given workstation. This is performed

using standard Microsoft Windows "hooks" that allow applications to intercept all keystrokes. The related data is then combined into the same recording as the explicit notifications. In effect, these provide the lowest level of granularity within the call flow recording, for example, appearing as:

- entering FirstName Determination Stage;
  - KeyHit="F1";
  - entering HelpScreen;
    - KeyHit="Esc";
  - leaving HelpScreen;
  - KeyHit="T";
  - KeyHit="o";
  - KeyHit="m"; and
- leaving FirstName Determination Stage.

The second means of inferring call progress relates to mouse